Software Engineering Project1b1

Stakeholders involved:

- Analytics
- Billing
- CRM Service
- Cart Service
- Catalog Service
- Charity Partner Service
- Compliance Service
- Customer
- Data Export
- Data Warehouse
- Dispatch Service
- Emergency Services
- Gamification Service
- Gift Service
- Inventory System
- Loyalty Service
- Maps Service
- Menu Service
- Notification Service
- Operations Console
- Order Management
- Order Management Service
- Order Tracking
- Payment Gateway
- Platform Admin
- Pricing Engine
- Profile Service
- Promo Engine
- Reputation Service
- Restaurant
- Restaurant System
- Rider
- Risk Engine
- Safety Service
- Subscription Service

- Sustainability Service
- Wallet Service

New Use Cases:

1. Group Ordering (Shared Cart)

Primary Actor: Customer

Supporting Actors: Cart Service, Payment Gateway, Notification Service

Preconditions: Customer logged in; restaurant selected; group invite link generated.

Postconditions:

- Success: Shared cart created, items added, and order placed.
- Failure: Cart not created or checkout failed.

Main Flow:

1. Customer selects Start Group Order.
2. System generates shareable invite link/code.
3. Participants join and add items.
4. Creator reviews items and places order.

Subflows:

- S1: Per-participant spending limits.
- S2: Each participant pays for own items.

Alternative Flows:

- A1: Participant timeout → items auto-removed.
- A2: Creator cancels group → participants notified.

2. Pre-Order for Pickup

Primary Actor: Customer

Supporting Actors: Restaurant System, Order Management

Preconditions: Customer logged in; pickup option available; time slots open.

Postconditions:

- Success: Order scheduled for pickup; confirmation issued.
- Failure: Pickup not scheduled.

Main Flow:

1. Customer selects Pickup option.
2. System displays available pickup windows.
3. Customer confirms time slot and places order.
4. System confirms pickup details and instructions.

Subflows:

- S1: Curbside pickup notes and vehicle details.

Alternative Flows:

- A1: Slot unavailable → suggest alternative time.

3. Wishlist / Favorites

Primary Actor: Customer

Supporting Actors: Catalog Service, Profile Service

Preconditions: Customer logged in.

Postconditions:

- Success: Items/restaurants saved.
- Failure: Favorites not saved.

Main Flow:

1. Customer taps Save to Favorites.
2. System stores item in profile.
3. Favorites accessible for quick order.

Subflows:

- S1: Smart reorder suggestions.

Alternative Flows:

- A1: Item unavailable → suggest similar.

4. Dietary & Nutrition Info Display

Primary Actor: Customer

Supporting Actors: Catalog Service, Compliance Service

Preconditions: Menu items have nutrition/allergen metadata.

Postconditions:

- Success: Nutrition and allergens displayed.
- Failure: Data unavailable.

Main Flow:

1. Customer opens item details.
2. System displays calories and allergen info.
3. Customer applies filters or flags allergens.

Subflows:

- S1: Warnings for selected allergens.

Alternative Flows:

- A1: Missing data → fallback message shown.

5. In-App Wallet (Top-up & Cashback)

Primary Actor: Customer

Supporting Actors: Wallet Service, Payment Gateway

Preconditions: Customer logged in; wallet enabled.

Postconditions:

- Success: Wallet topped up and usable.
- Failure: Wallet operation fails.

Main Flow:

1. Customer opens Wallet and selects top-up.
2. Payment Gateway processes top-up.
3. Wallet balance updates and is usable at checkout.

Subflows:

- S1: Cashback accrual on eligible orders.

Alternative Flows:

- A1: Payment failure → retry or alternate method.

6. Gift Cards / Vouchers (Purchase & Redemption)
Primary Actor: Customer

Supporting Actors: Gift Service, Payment Gateway, Notification Service

Preconditions: Customer logged in.

Postconditions:

- Success: Gift card purchased or redeemed.
- Failure: Transaction fails.

Main Flow:

1. Customer selects Buy Gift Card or Redeem.
2. System processes payment or validates code.
3. Balance added to account for future use.

Subflows:

- S1: Send gift via email/SMS.

Alternative Flows:

- A1: Invalid/expired code → error message.

7. Subscription / Membership for Free Delivery
Primary Actor: Customer

Supporting Actors: Subscription Service, Payment Gateway

Preconditions: Customer logged in; membership available.

Postconditions:

- Success: Membership activated with benefits.
- Failure: Membership not activated.

Main Flow:

1. Customer selects membership plan.
2. System processes recurring payment.
3. Delivery benefits auto-applied at checkout.

Subflows:

- S1: Trial period with auto-renewal.

Alternative Flows:

- A1: Payment failure → membership paused.

8. Shareable Tracking Links

Primary Actor: Customer

Supporting Actors: Order Tracking, Notification Service

Preconditions: Active order in progress.

Postconditions:

- Success: Shareable link created.
- Failure: Link not generated.

Main Flow:

1. Customer taps Share Tracking.
2. System generates secure tracking link.
3. Recipient views ETA and driver location.

Subflows:

- S1: Link expiry or revoke access.

Alternative Flows:

- A1: Privacy settings block sharing.

9. Gamification (Badges & Streaks)

Primary Actor: Customer

Supporting Actors: Gamification Service, Notification Service

Preconditions: Customer logged in.

Postconditions:

- Success: Badges awarded; streaks tracked.
- Failure: No rewards assigned.

Main Flow:

1. System tracks qualifying actions.
2. Customer earns badges/streaks.
3. Rewards unlocked at thresholds.

Subflows:

- S1: Leaderboards among friends.

Alternative Flows:

- A1: Streak broken → grace period applied.

10. Carbon Footprint Display

Primary Actor: Customer

Supporting Actors: Sustainability Service, Catalog Service

Preconditions: Emissions data available per item.

Postconditions:

- Success: Footprint displayed.
- Failure: Data unavailable.

Main Flow:

1. Customer views item or cart.
2. System calculates estimated CO_2e.
3. Eco-friendly alternatives highlighted.

Subflows:

- S1: Opt-in for low-carbon default filter.

Alternative Flows:

- A1: Missing data → estimate unavailable.

11. Dynamic Pricing (Restaurant-Side Controls)
Primary Actor: Restaurant

Supporting Actors: Pricing Engine, Catalog Service

Preconditions: Restaurant onboarded; rules configured.

Postconditions:

- Success: Dynamic prices applied.
- Failure: Pricing rules not applied.

Main Flow:

1. Restaurant sets peak/off-peak rules.
2. System updates menu prices.
3. Customers see updated prices.

Subflows:

- S1: Automatic promotions in off-peak slots.

Alternative Flows:

- A1: Conflict with platform promos → resolve.

12. Promotion Campaign Creation (Restaurant Self-Serve)
Primary Actor: Restaurant

Supporting Actors: Promo Engine, Analytics

Preconditions: Restaurant logged in; budget set.

Postconditions:

- Success: Campaign launched.
- Failure: Campaign rejected or paused.

Main Flow:

1. Restaurant defines offer and rules.
2. System validates constraints.
3. Campaign published to users.

Subflows:

- S1: A/B test creatives and placements.

Alternative Flows:

- A1: Low performance → auto-optimize.

13. Inventory Management Integration

Primary Actor: Restaurant

Supporting Actors: Inventory System, Catalog Service

Preconditions: Inventory integration enabled.

Postconditions:

- Success: Out-of-stock items hidden.
- Failure: Inventory sync fails.

Main Flow:

1. System syncs inventory data.
2. Unavailable items hidden from menu.
3. Low-stock alerts sent to restaurant.

Subflows:

- S1: Auto-substitute suggestions.

Alternative Flows:

- A1: Sync outage → fallback to manual toggle.

14. Customer Segmentation (Restaurant View)

Primary Actor: Restaurant

Supporting Actors: Analytics, CRM Service

Preconditions: Sufficient order history/consent.

Postconditions:

- Success: Segments created.
- Failure: No usable segments.

Main Flow:

1. Restaurant selects segmentation criteria.
2. System builds segments and insights.
3. Restaurant targets promos to segments.

Subflows:

- S1: Lookalike audiences from top customers.

Alternative Flows:

- A1: Privacy opt-outs → users excluded.

15. Meal Bundle/Combos Configuration

Primary Actor: Restaurant

Supporting Actors: Menu Service, Pricing Engine

Preconditions: Restaurant logged in; bundle defined.

Postconditions:

- Success: Bundle published.
- Failure: Bundle not published.

Main Flow:

1. Restaurant creates bundle template.
2. System validates dependencies.
3. Bundle appears in menu with upsell logic.

Subflows:

- S1: Time-limited availability.

Alternative Flows:

- A1: Item out-of-stock → auto-adjust bundle.

16. Heatmap of High-Demand Areas

Primary Actor: Rider

Supporting Actors: Dispatch Service, Maps Service

Preconditions: Rider online; location services enabled.

Postconditions:

- Success: Rider navigates to hotspot.
- Failure: Heatmap unavailable.

Main Flow:

1. Rider opens heatmap view.
2. System highlights demand zones.
3. Rider heads to selected zone.

Subflows:

- S1: Incentives for zone presence.

Alternative Flows:

- A1: Data stale → refresh prompt.

17. Earnings Dashboard & Analytics

Primary Actor: Rider

Supporting Actors: Wallet Service, Analytics

Preconditions: Rider logged in; earnings data available.

Postconditions:

- Success: Earnings insights shown.
- Failure: Data not displayed.

Main Flow:

1. Rider opens earnings dashboard.

2. System shows daily/weekly metrics.
3. Trends and insights displayed.

Subflows:

- S1: Tax summary download.

Alternative Flows:

- A1: Data delay → last known values.

18. Instant Cash-Out of Earnings

Primary Actor: Rider

Supporting Actors: Wallet Service, Payment Gateway

Preconditions: Rider has balance and payout method.

Postconditions:

- Success: Payout completed.
- Failure: Cash-out declined.

Main Flow:

1. Rider selects Instant Cash-Out.
2. System validates limits and fees.
3. Funds transferred with confirmation.

Subflows:

- S1: Daily/weekly payout limits.

Alternative Flows:

- A1: Risk hold → standard payout.

19. Rider Safety Check-In (SOS)

Primary Actor: Rider

Supporting Actors: Safety Service, Emergency Services

Preconditions: Rider online; trip active.

Postconditions:

- Success: Safety status confirmed.
- Failure: Check-in not completed.

Main Flow:

1. Rider taps Safety Check-In.
2. System prompts status update.
3. If unsafe, SOS triggers escalation.

Subflows:

- S1: Auto check-ins in flagged zones.

Alternative Flows:

- A1: No response → auto escalation.

20. Rider Rating & Feedback on Customers

Primary Actor: Rider

Supporting Actors: Reputation Service

Preconditions: Completed delivery.

Postconditions:

- Success: Feedback stored.
- Failure: Feedback not saved.

Main Flow:

1. Rider rates customer post-delivery.
2. System captures feedback tags.
3. Insights used for quality control.

Subflows:

- S1: Customer coaching if patterns emerge.

Alternative Flows:

- A1: Abuse detected → moderation.

21. Fraud Detection & Alerts
Primary Actor: Platform Admin

Supporting Actors: Risk Engine, Payment Gateway

Preconditions: Live orders and payment events.

Postconditions:

- Success: Suspicious activity flagged.
- Failure: Fraud not detected.

Main Flow:

1. System runs fraud checks.
2. High-risk orders challenged.
3. Admins receive alerts.

Subflows:

- S1: Step-up verification (ID check).

Alternative Flows:

- A1: False positive → release order.

22. System-Wide Promotions & Campaigns
Primary Actor: Platform Admin

Supporting Actors: Promo Engine, Analytics

Preconditions: Budget and targeting defined.

Postconditions:

- Success: Campaign launched.
- Failure: Campaign not launched.

Main Flow:

1. Admin configures promo rules.
2. System validates constraints.
3. Campaign goes live.

Subflows:

- S1: Auto-optimization towards goals.

Alternative Flows:

- A1: Overspend risk → throttle campaign.

23. Service Availability Toggling (Geo-Pause)
Primary Actor: Platform Admin

Supporting Actors: Operations Console, Maps Service

Preconditions: Operational incident detected.

Postconditions:

- Success: Deliveries paused/resumed.
- Failure: Toggle not applied.

Main Flow:

1. Admin selects geographic zones.
2. System pauses deliveries in zone.
3. Customer UI updated.

Subflows:

- S1: Automated toggles via alerts.

Alternative Flows:

- A1: Merchant exceptions allowed.

24. Advanced Analytics Dashboard
Primary Actor: Platform Admin

Supporting Actors: Analytics, Data Warehouse

Preconditions: Data pipeline healthy.

Postconditions:

- Success: KPIs displayed.
- Failure: No analytics rendered.

Main Flow:

1. Admin opens analytics dashboard.
2. System shows sales and conversion KPIs.
3. Drill-down by cohort and region.

Subflows:

- S1: Scheduled reports and alerts.

Alternative Flows:

- A1: Data latency → stale data warning.

25. Regulatory Compliance Tools

Primary Actor: Platform Admin

Supporting Actors: Compliance Service, Billing, Data Export

Preconditions: Jurisdictional rules configured.

Postconditions:

- Success: Tax invoices generated, GDPR exports processed.
- Failure: Compliance task fails.

Main Flow:

1. Admin triggers invoice or export.
2. System produces compliant docs.
3. Admin delivers reports as required.

Subflows:

- S1: Data retention and erasure workflows.

Alternative Flows:

- A1: Identity not verified → request rejected.

26. Donation Option (Round Up to Charity)

Primary Actor: Customer

Supporting Actors: Payment Gateway, Charity Partner Service

Preconditions: Customer logged in; charity partners available.

Postconditions:

- Success: Donation processed.
- Failure: Donation not applied.

Main Flow:

1. Customer enables Round Up at checkout.
2. System calculates donation amount.
3. Donation processed with order.

Subflows:

- S1: Customer selects preferred charity.

Alternative Flows:

- A1: Charity service unavailable.

27. Promo Code Application & Validation
Primary Actor: Customer

Supporting Actors: Promo Engine, Payment Gateway, Notification Service

Preconditions: Customer logged in; items in cart; promo code entered.

Postconditions:

- Success: Promo applied, discount reflected in cart.
- Failure: Promo rejected; error message shown.

Main Flow:

1. Customer enters promo code at checkout.
2. System validates promo eligibility.
3. System applies discount if valid.
4. Cart total updates; confirmation displayed.

Subflows:

- S1: Auto-apply best available promo.

- S2: Suggest alternate promo if entered code expired.

Alternative Flows:

- A1: Invalid/expired promo → error message.
- A2: Promo conditions not met (min spend, restaurant-specific).
- A3: System outage → skip promo validation.

28. Order Scheduling (Future Delivery)
Primary Actor: Customer

Supporting Actors: Order Management Service, Notification Service

Preconditions: Customer logged in; items in cart; future slot available.

Postconditions:

- Success: Order scheduled with delivery time.
- Failure: Order not scheduled.

Main Flow:

1. Customer selects 'Schedule Delivery'.
2. System shows available slots.
3. Customer selects date and time.
4. System reserves slot and confirms order.

Subflows:

- S1: Smart recommendation of least busy slots.

Alternative Flows:

- A1: Slot unavailable → prompt to select another.
- A2: Restaurant closed on selected time → suggest alternatives.

29. Loyalty Points Redemption
Primary Actor: Customer

Supporting Actors: Loyalty Service, Payment Gateway

Preconditions: Customer logged in; account has loyalty points.

Postconditions:

- Success: Points redeemed and applied to bill.
- Failure: No redemption applied.

Main Flow:

1. Customer opts to pay with loyalty points.
2. System checks available balance.
3. Points are applied as discount.
4. Order total updates accordingly.

Subflows:

- S1: Auto-suggest redemption if sufficient points available.

Alternative Flows:

- A1: Insufficient points → prompt to earn more.
- A2: Loyalty service unavailable → fallback to normal payment.

30. Multi-Restaurant Cart
Primary Actor: Customer

Supporting Actors: Cart Service, Order Management

Preconditions: Customer logged in; browsing menus.

Postconditions:

- Success: Items from multiple restaurants saved in cart.
- Failure: Cart restricted to one restaurant.

Main Flow:

1. Customer adds items from Restaurant A.
2. Customer adds items from Restaurant B.
3. System splits cart into separate orders.
4. System calculates delivery fees per restaurant.

Subflows:

- S1: Suggest restaurants that allow bundling.

Alternative Flows:

- A1: Policy restricts cart to one restaurant → system prompts warning.

31. Tip Rider During Checkout
Primary Actor: Customer

Supporting Actors: Payment Gateway, Rider Wallet

Preconditions: Customer logged in; order ready for payment.

Postconditions:

- Success: Tip added and recorded for rider.
- Failure: Tip not added.

Main Flow:

1. Customer selects tip amount during checkout.
2. System updates order total.
3. Payment Gateway processes combined payment.
4. Rider wallet updated with tip.

Subflows:

- S1: Suggest default tip percentage options.

Alternative Flows:

- A1: Customer skips tipping option.

32. Push Notifications Management
Primary Actor: Customer

Supporting Actors: Notification Service

Preconditions: Customer logged in; app installed on device.

Postconditions:

- Success: Notifications delivered or turned off.
- Failure: Notifications not delivered.

Main Flow:

1. Customer enables notifications in settings.
2. System pushes real-time updates for orders and promos.
3. Customer manages preferences (mute, categories).

Subflows:

- S1: Silent notifications for background updates.

Alternative Flows:

- A1: Customer blocks notifications at device level.
- A2: Notification service outage → fallback to SMS/email.

33. In-App Chat with Rider
Primary Actor: Customer

Supporting Actors: Chat Service, Rider

Preconditions: Active order with assigned rider.

Postconditions:

- Success: Customer and rider exchange messages.
- Failure: Messages not delivered.

Main Flow:

1. Customer opens chat from order details.
2. System connects chat session with assigned rider.
3. Messages exchanged securely within app.

Subflows:

- S1: Auto-translate messages if languages differ.

Alternative Flows:

- A1: Rider unresponsive → escalate to support.
- A2: Chat service unavailable → fallback to call.

34. Order Gift to Another User
Primary Actor: Customer

Supporting Actors: Gift Service, Payment Gateway, Notification Service

Preconditions: Customer logged in; recipient address provided.

Postconditions:

- Success: Order placed as a gift; recipient notified.
- Failure: Gift not processed.

Main Flow:

1. Customer selects 'Send as Gift'.
2. System prompts for recipient details and message.
3. Customer confirms order and payment.
4. Recipient receives notification or gift card.

Subflows:

- S1: Gift scheduling for birthdays/occasions.

Alternative Flows:

- A1: Invalid recipient address → error message.

35. Voice Search for Food Items
Primary Actor: Customer

Supporting Actors: Voice Recognition Service, Catalog Service

Preconditions: Customer logged in; microphone access granted.

Postconditions:

- Success: Voice command processed into search results.
- Failure: Voice not recognized.

Main Flow:

1. Customer activates voice search.
2. Customer speaks food name or cuisine.
3. System transcribes and fetches results.
4. Results displayed on app.

Subflows:

- S1: Suggestions shown while speaking.

Alternative Flows:

- A1: Background noise prevents recognition.
- A2: No match found → system suggests similar items.

Differences seen in the LLM reports:

Gemini Pro - Through the RAG code, Gemini did generate the chunks as the input to be passed into the LLM, but the final result itself was not sufficient. It could not generate precise use cases with respect to a food delivery app, even after referring to the PDFs fed to it. It gave very generic output.

Llama3.2 - Llama3.2 was the second best model that we used. It perfectly understood the prompt, carefully created the chunks and then gave the use cases as required. The only shortcoming was that it was not able to generate clear use cases with the preconditions, postconditions, main flow, subflows and alternate flows. It could efficiently give one line use cases which were valid too, wrt a food delivery system.

Deepseek - We used Deepseek's deepseek-r1:latest model which was running locally. It was better than both Gemini Pro and Llama; with some carefully written prompts it gave accurate use cases. It often mentioned that the query lacks information and that it won't be able to give any proper answers. It requires careful prompting. After giving some examples it was able to generate use cases. But to get new, useful, and logically tight use cases, very specific prompts were required.

Total cost of LLM usage:

As of now, we haven't bought any subscriptions for any of the LLMs, hence, the cost is $0. We have free Gemini Pro through university credentials, which we used for RAG, as one of the models.